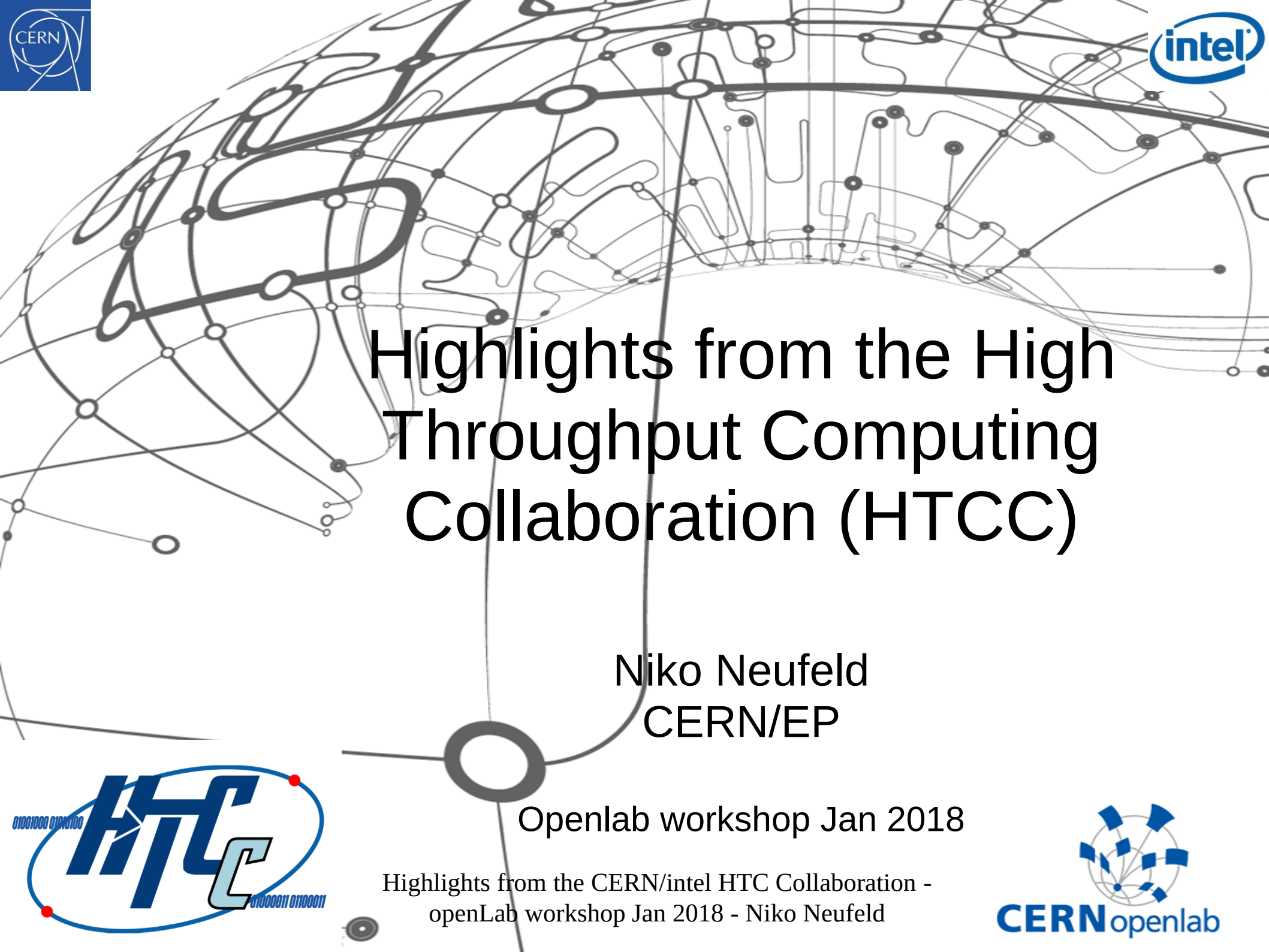


Highlights from the High Throughput Computing Collaboration (HTCC)

Niko Neufeld
CERN/EP

Openlab workshop Jan 2018

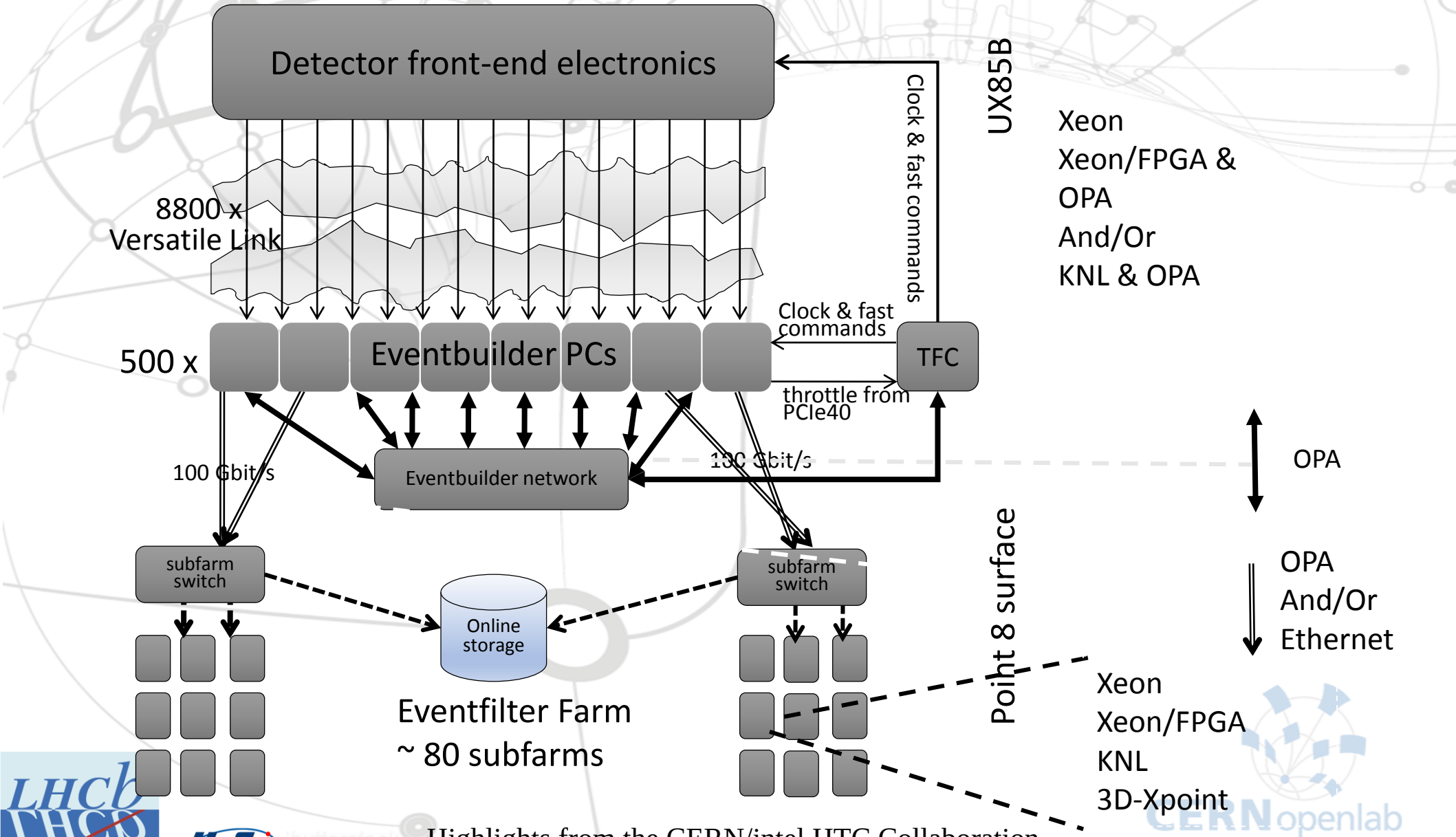
Summary slide from SC14 (still valid ☺)

- The LHC experiments need to reduce 100 TB/s to ~ 25 PB/ year
- Today this is achieved with massive use of custom ASICs and in-house built FPGA-boards and x86 computing power
- Finding new physics requires massive increase of processing power, much more flexible algorithms in software and much faster interconnects
- The CERN/Intel HTC Collaboration will explore Intel's Xeon/FPGA concept, XeonPhi and OmniPath technologies to build the LHC trigger/DAQ of the future

Intel technologies in HTCC

- OmniPath
- Xeon/FPGA
- Xeon/Phi
- QAT
- NVMeoF

LHC TDAQ architecture using Intel

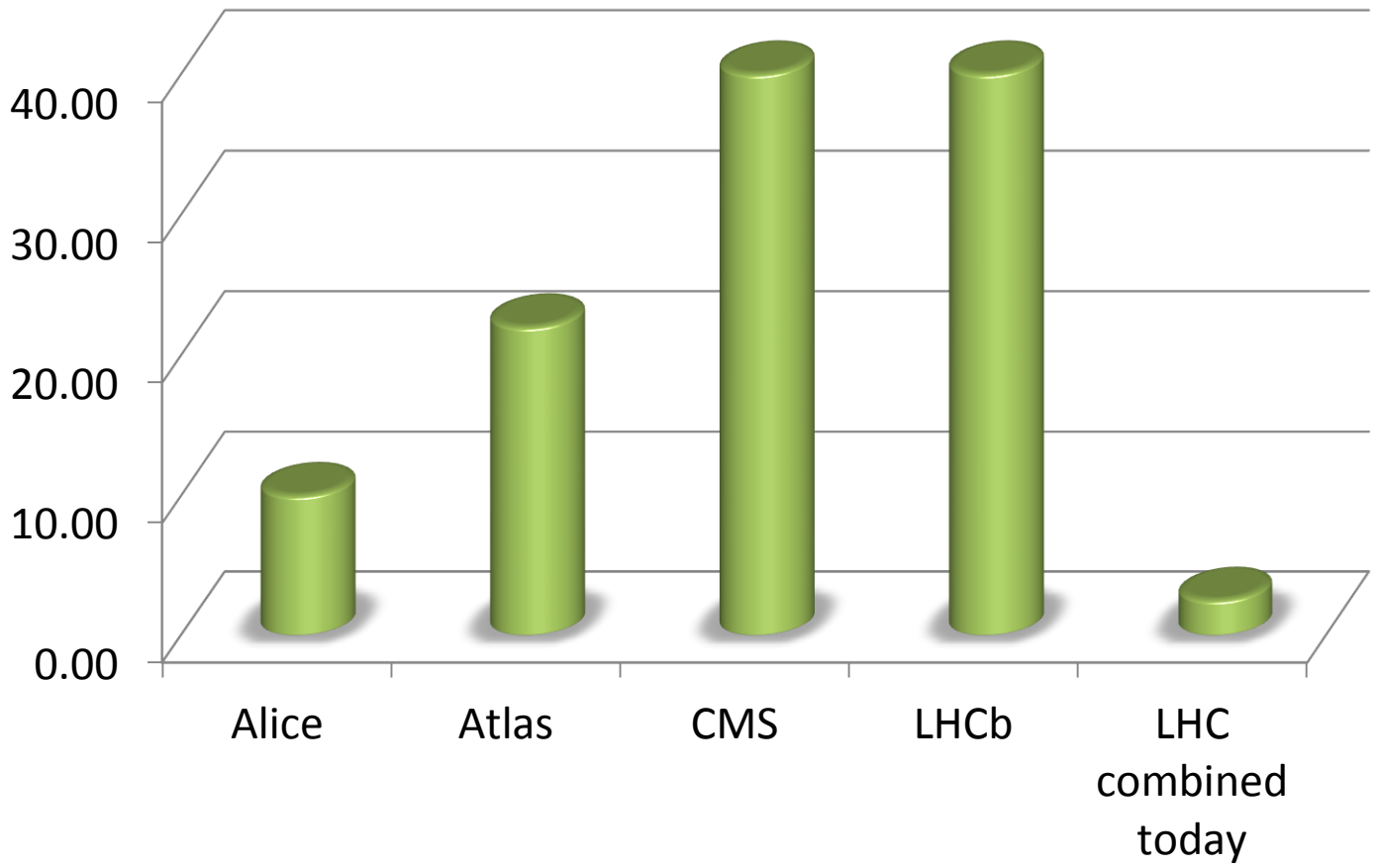


DAQ challenge

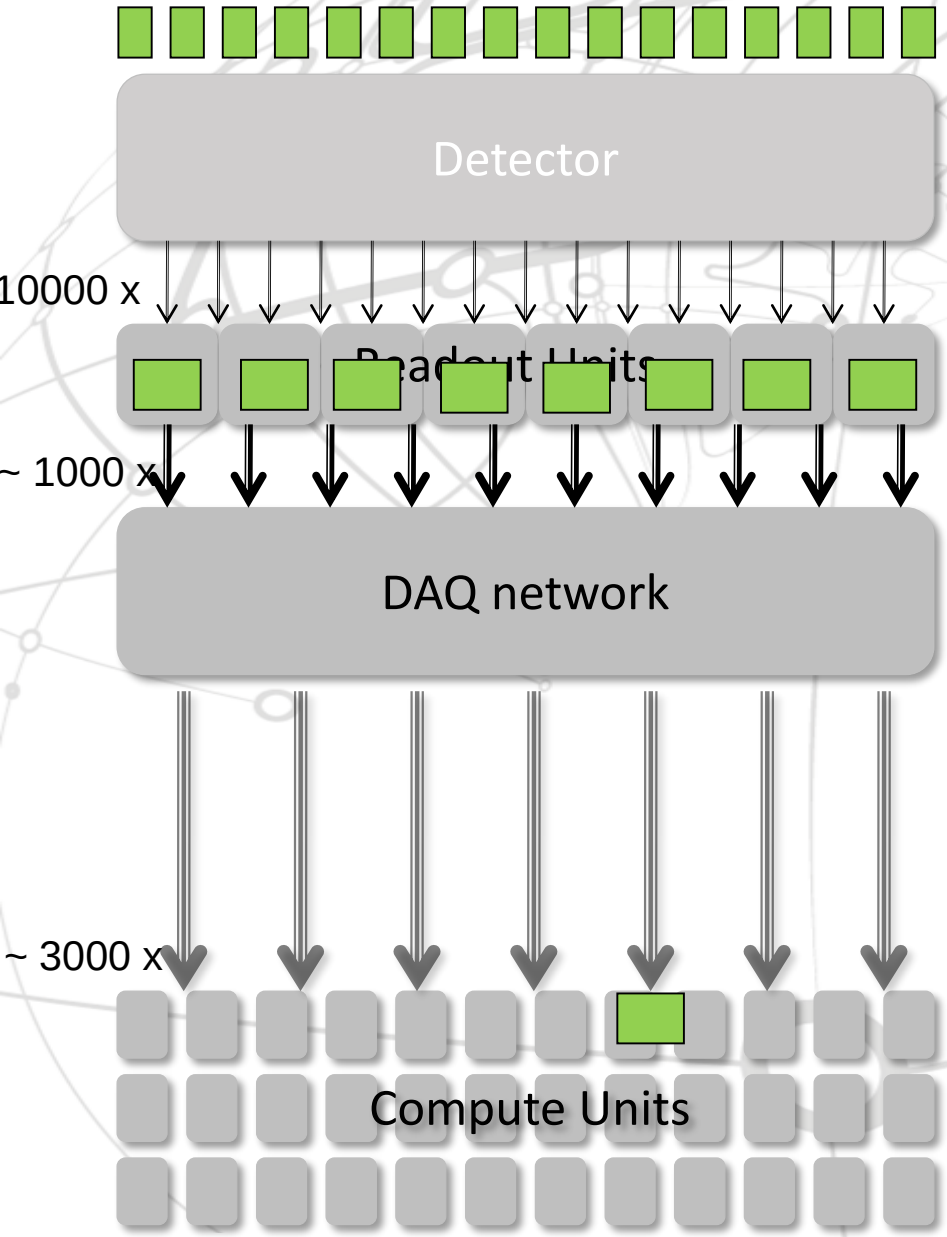
- Transport multiple Terabit/s reliably and cost-effectively
- Integrate the network closely and efficiently with compute resources (be they classical CPU or “many-core”)
- Multiple network technologies should seamlessly co-exist in the same integrated fabric (“the right link for the right task”)

A 40 Tbit/s network

Network – Projected Throughput [Tbit/s]



Event-building for the LHC (from Run3)



- Pieces of collision data spread out over 10000 links
- All pieces must be brought together into one of thousands compute units
- Compute units running complex filter algorithms (today dual-socket Xeon servers)

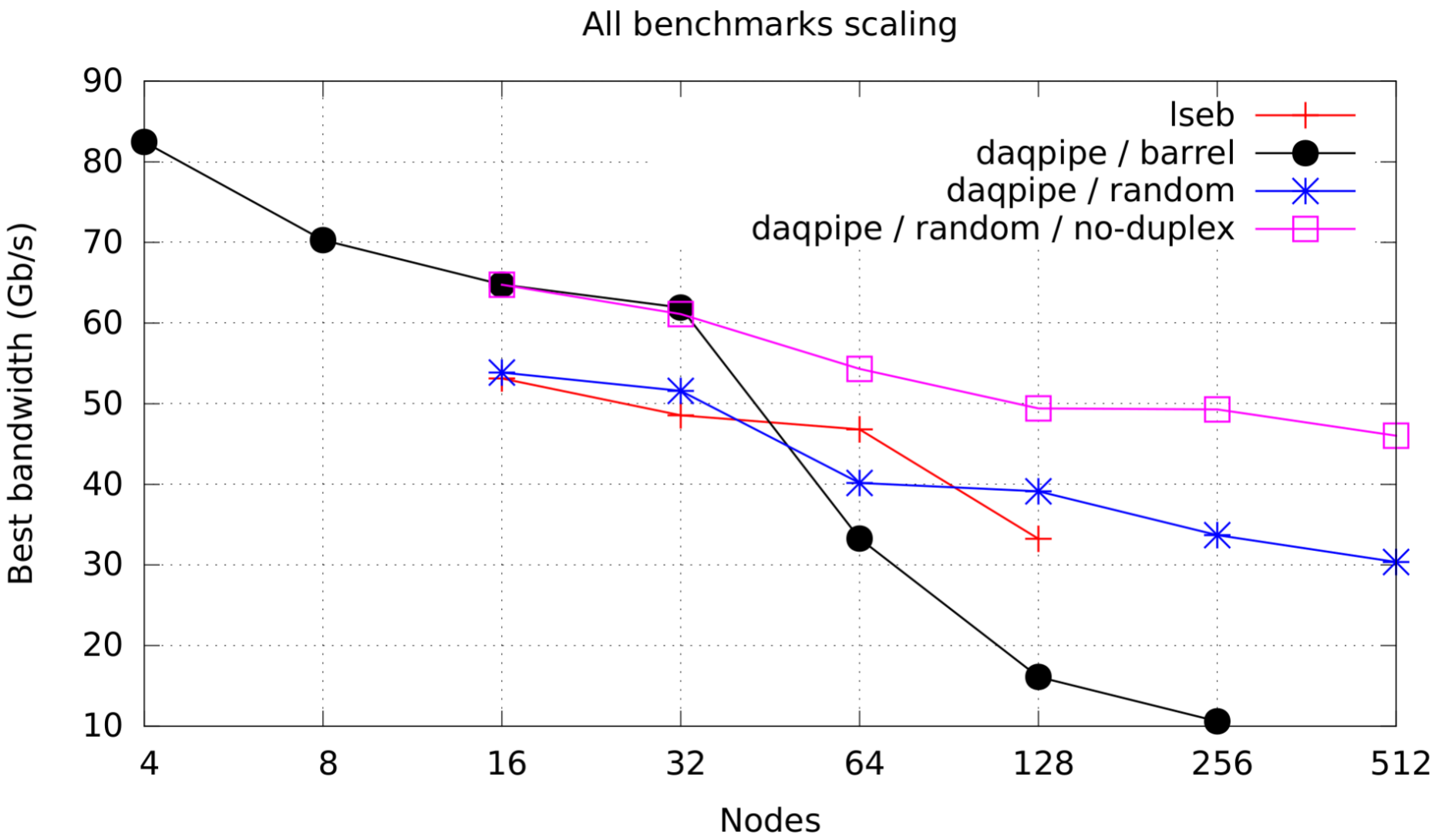
custom radiation- hard link from the detector 3.2 / 4.8 Gbit/s (CERN Versatile Link VL)

DAQ (“event-building”) links – some LAN (10/40/100 Gbit/s)

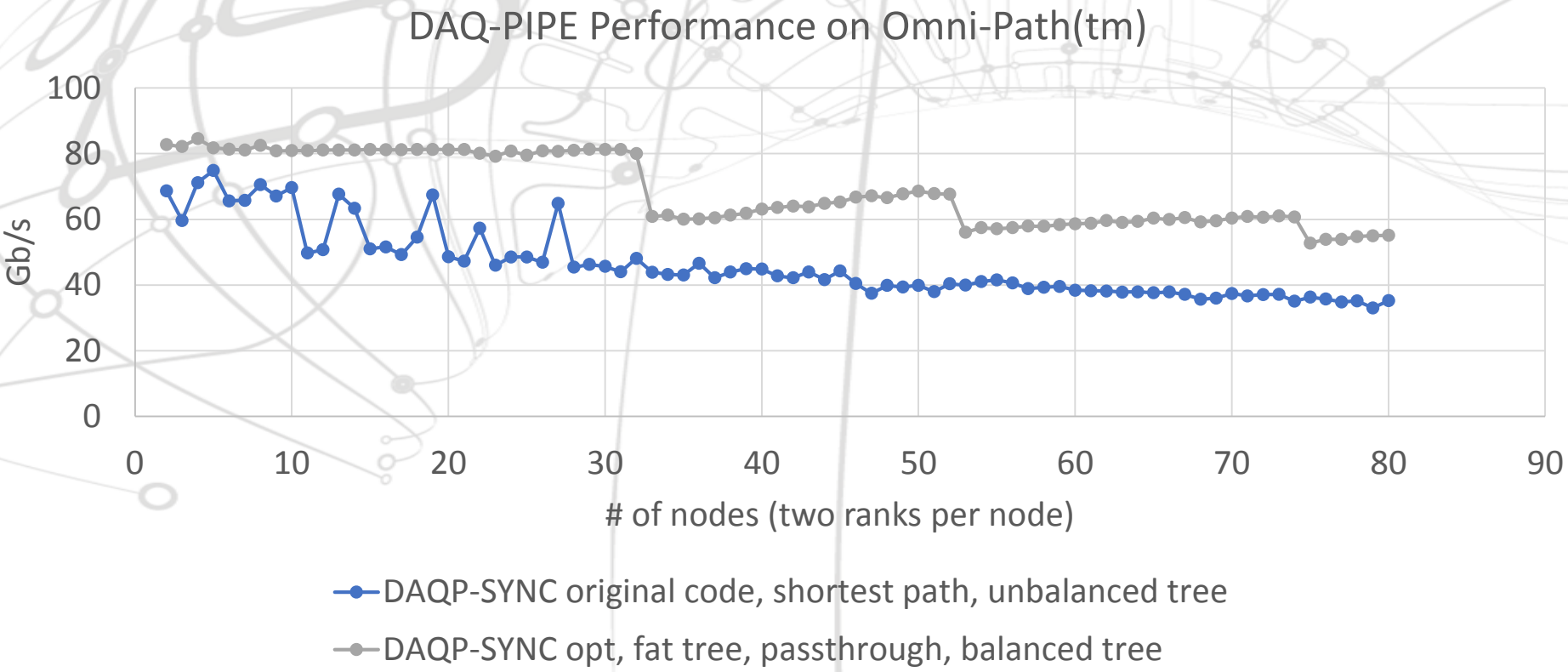
How to test a 40 Tbit/s system without buying one?

- Answer: there are quite a few such systems out there. Just check the Top500!
- We have developed a highly portable software package (DAQPIPE) which can completely emulate such a data-acquisition system on an HPC site
  - It supports multiple protocols and network technologies and allows one to scan for extensive combinations of ***many relevant parameters*** (message size/rate, buffers, push/pull, scheduling etc…)

Scaling on Marconi super-computer (Omni-Path)



Scaling Eventbuilding on OPA

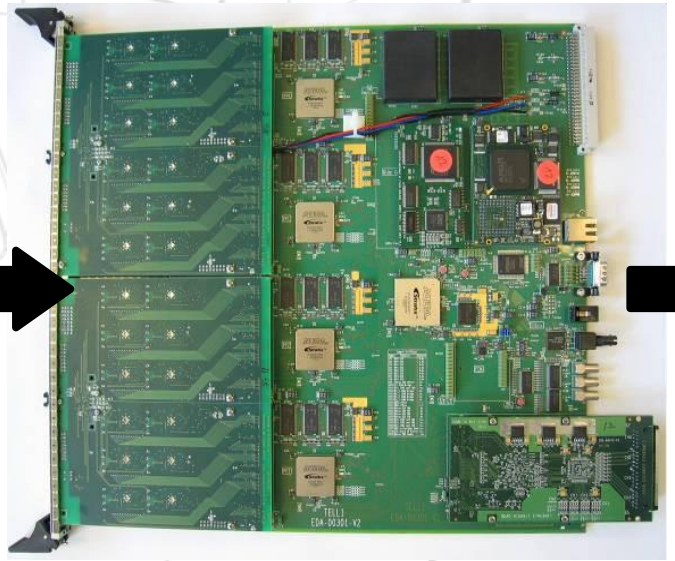


- Scaling the DAQ network to more nodes reveals performance issues
- HTCC finds much better tuning for OmniPath and adaptation of event-building to RDMA
- Performance can still be improved (better balancing) → new tests scheduled.

General Readout Chain

Optical links

Fast networks



Readout electronic
for detectors
(Custom)

Mainly ASICs
In low rad. areas

FPGAs

Back-end electronics
(Custom)

Many FPGAs and CCPCs

Computing farms
(Commercial)

FPGA usage to be investigat

Test case: LHCb Calorimeter Raw Data Decoding

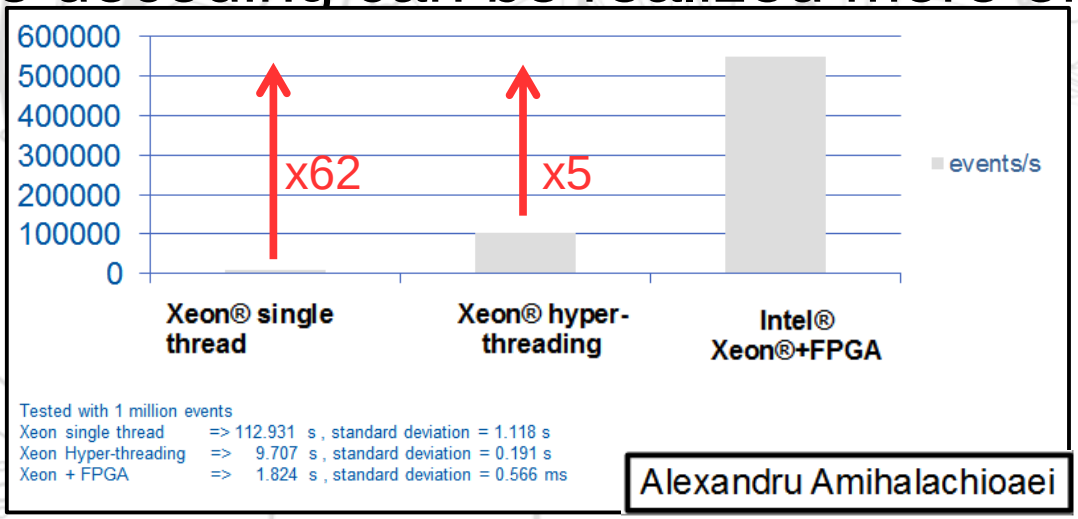
- Two types of calorimeters in LHCb: ECAL/HCAL
- 32 ADC channels for each FEB of 238 FEBs
- Raw data format:
  - ADC data is sent using 4 bits or 12 bits

LHCb Calorimeter raw data bank

Control word (9b) (Figure 18)	Crate (5b)	Card (4b)	Length ADC (7b)	Length trigger (7b)
Trigger bit pattern (32b)				
Zero padding	Trigger (8b)	Trigger (8b)	Trigger (8b)	
ADC bit pattern (32b)				
ADC low	ADC long (12b)	ADC long (12b)	ADC (4b)	
Zero padding at the end	ADC long (12b)	ADC high (8b)		

Results Calorimeter Raw Data Decoding: Ivy Bridge + StratixV

- On FPGAs the decoding can be realized more efficiently



Measured

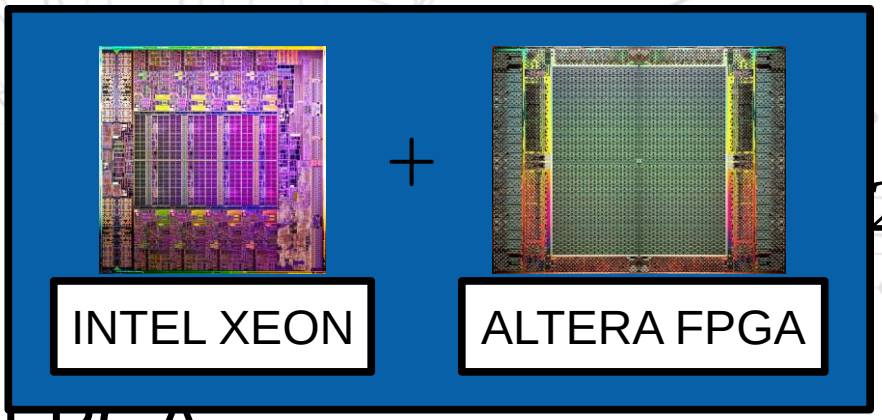
- Bottleneck is bandwidth between CPU and FPGA

add more cores, tested BDW + Arria10 GX FPGA

FPGA resources:

FPGA Resource Type	FPGA Resources used [%]	For Interface used [%]
ALMs	58	30
DSPs	0	0
Registers	15	5

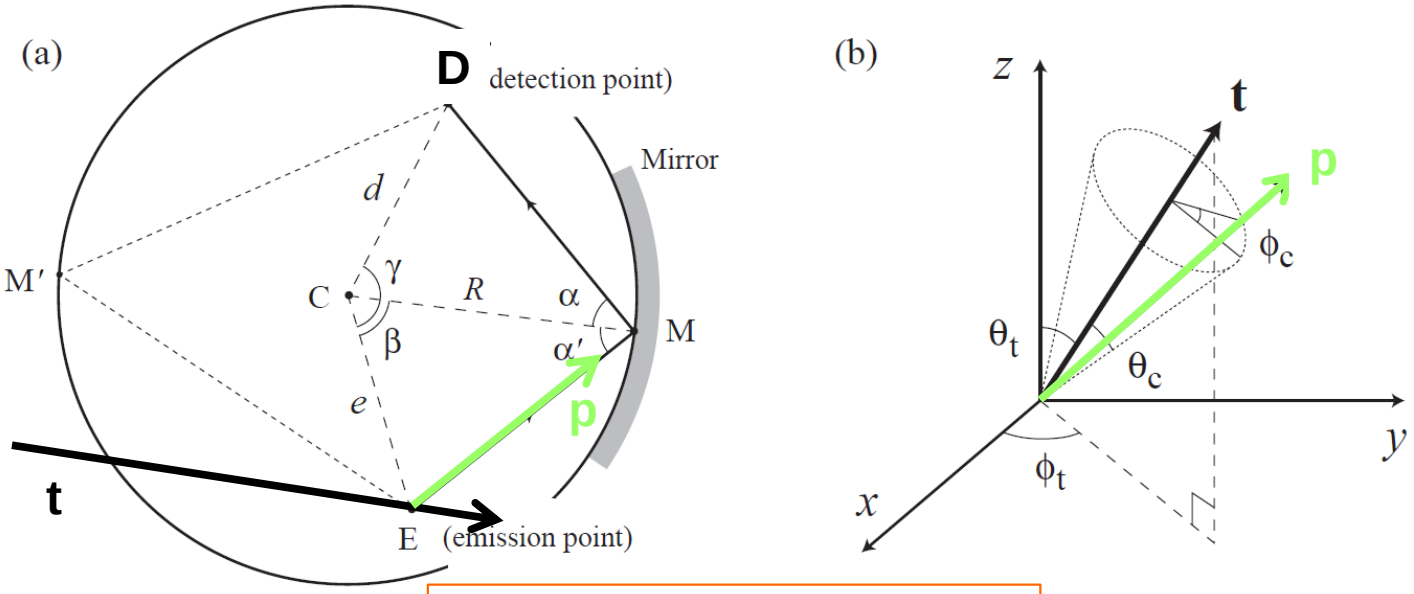
Intel Xeon/FPGA



- Two socket system:
- First: Intel(R) Xeon(R) 26 v2
- Second: Altera Stratix V GX A7 FPGA
- 234'720 ALMs, 940'000 Registers, 256 DSPs
- Host Interface: high-bandwidth and low latency
- Memory: Cache-coherent access to main memory
- Programming model : Verilog and OpenCL

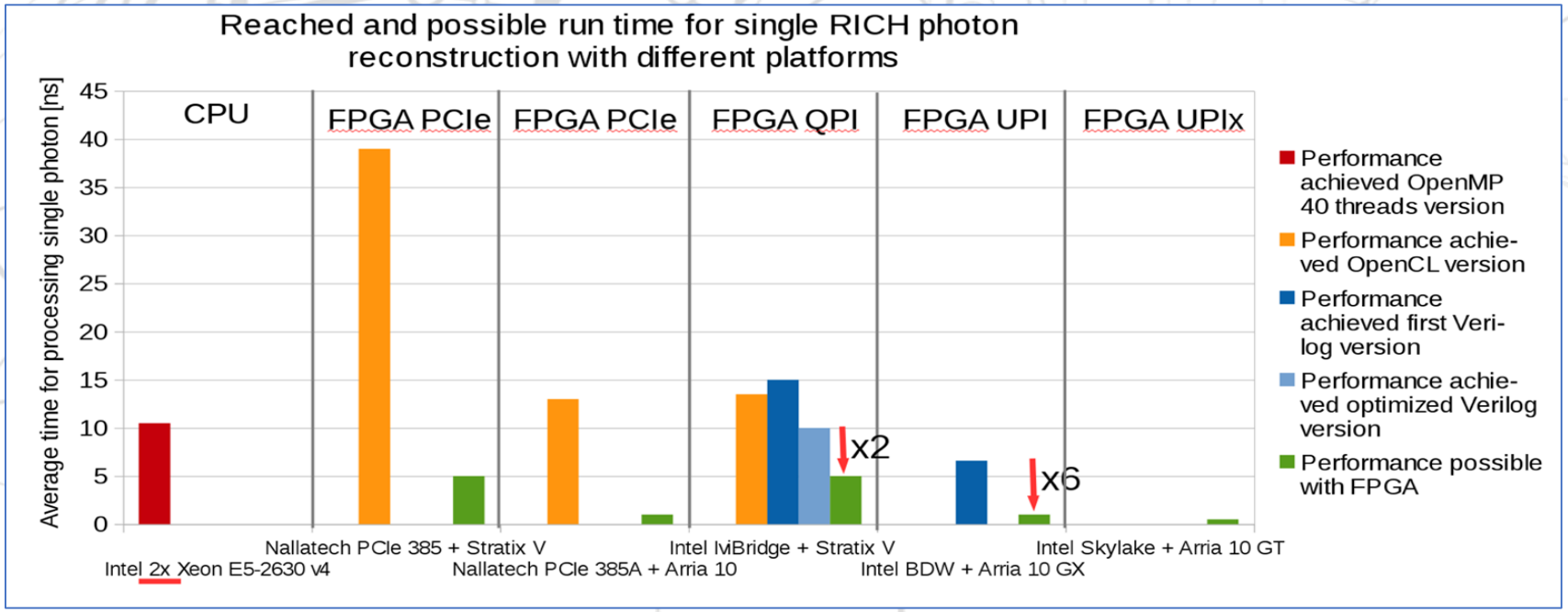
Test case: RICH PID Algorithm

- Calculate Cherenkov angle Θ_c for each track **t** and detection point **D**
- RICH PID is not processed for every event, processing time too long!



Reference: LHCb Note LHCb-98-040

Performance for photon-finding



- Lower is better ☺
- In particular in power-consumption, see next slide

Compare Verilog - OpenCL

- Development time
  - 2.5 months – 2 weeks — **Faster**
  - 3400 lines Verilog – 250 lines C — **Easier**
- Performance
  - CQRT: x35 – x30
  - RICH: x35 – x26

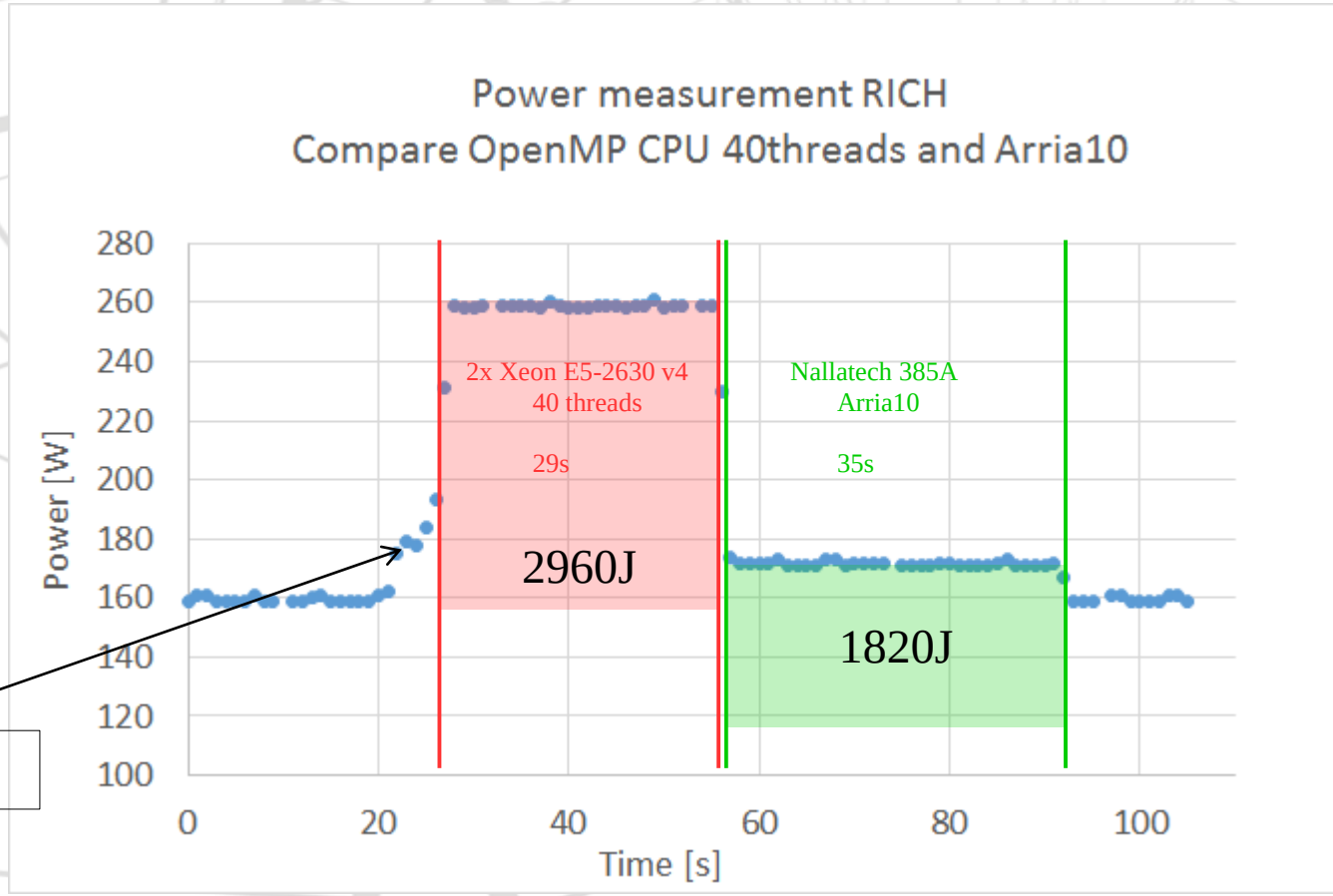
  Similar performance
- FPGA resource usage

RICH Kernel

FPGA Resource Type	Verilog RTL: FPGA Resources used [%]	OpenCL: FPGA Resources used [%]
ALMs	88	63
DSPs	67	82
Registers	48	24

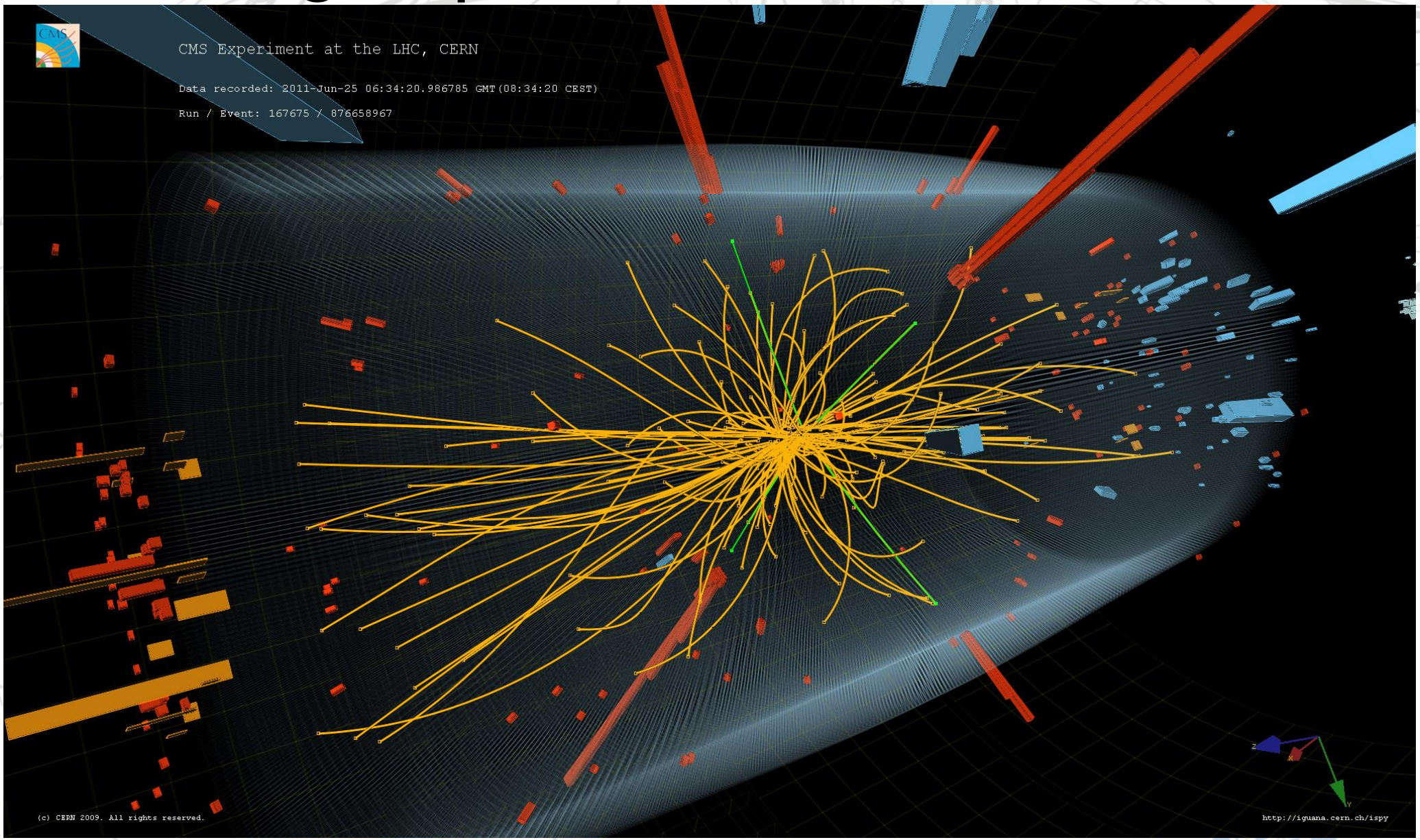
Similar resource usage

“Greener” photons

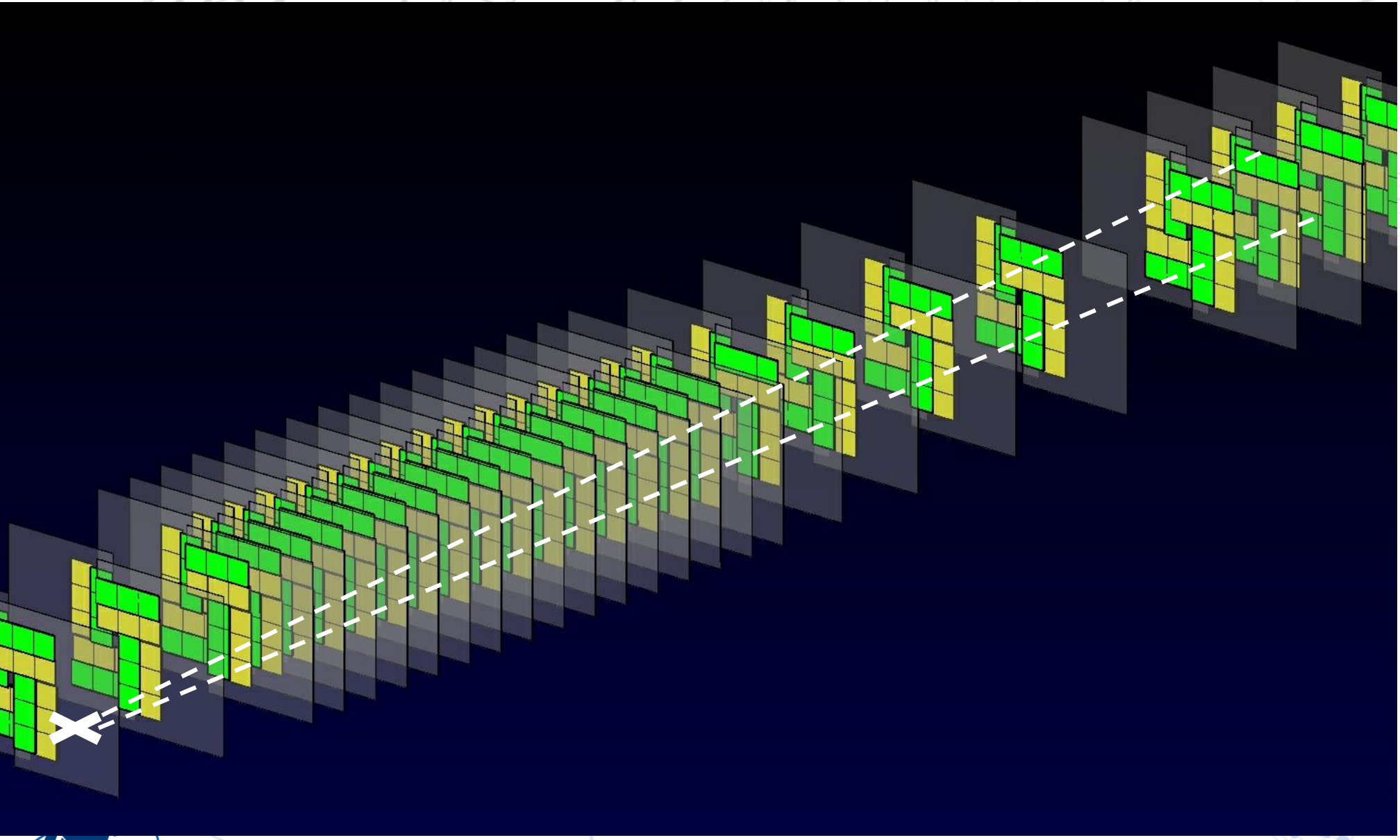


16777216 random photons
Multi loop factor: 160
Used CPU threads: 40, well vectorized
FPGA BSP not optimized (high idle-power, more improvements possible)

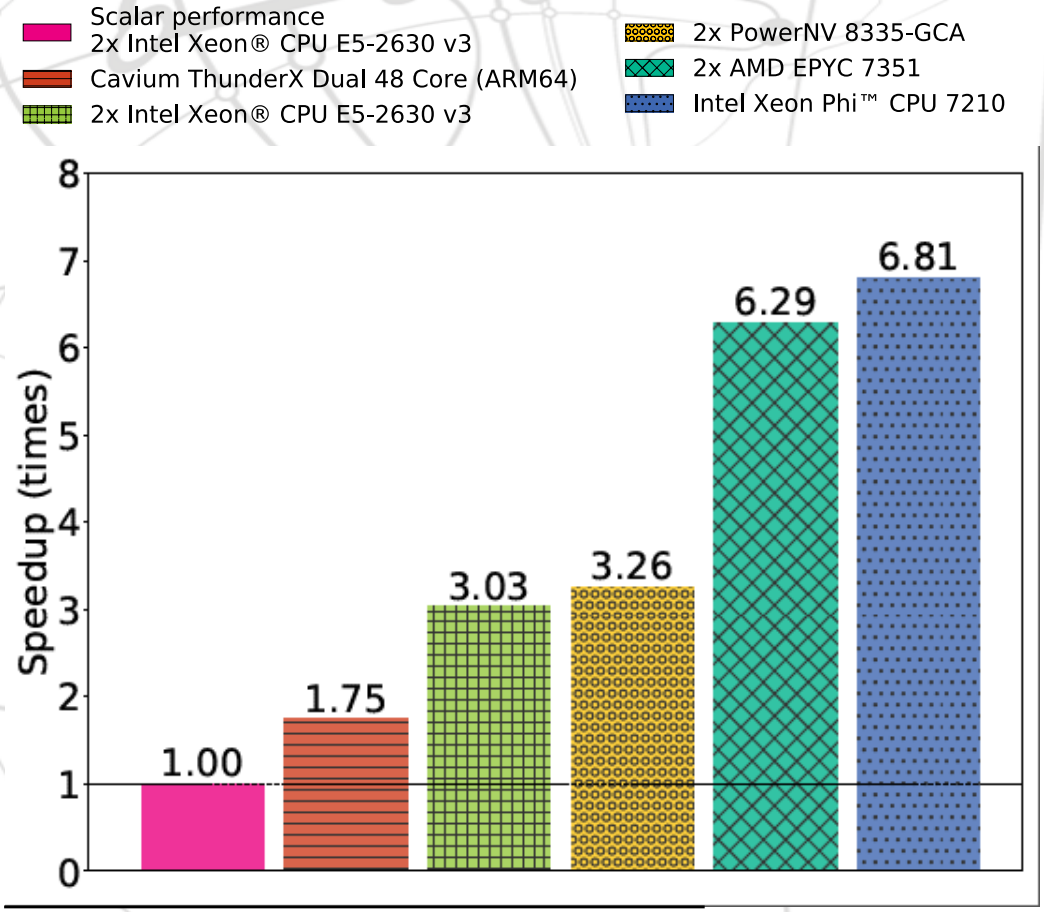
Charged particles in the detector



Tracking those particles…



Vectorized & threaded tracking in LHCb



- Detailed analysis and roofline models done
- KNL looks very attractive for this problem
- Even more so when putting in list-price for CPUs, where known (but we don't talk about money in openlab so I refrain ☺)

Summary

- Over the past 3 years HTCC has been evaluating upcoming Intel technologies for their potential use in LHC Trigger & DAQ systems
- Building very high bandwidth DAQ networks using OmniPath is feasible!
- Vectorized and parallelized code for important trigger algorithms can be run (very) efficiently on Xeon/Phi / KNL (lighter-weight cores with many vector engines)
- FPGAs are now a viable alternative also for complex algorithms, they are power-efficient and their programming model is now not necessarily more complicated than other acceleration frameworks (OpenCL, CuDA or OpenACC)
- Hardware compression can provide nice saving ”for free” (Intel Quick Assist) at enormous rates